Structured Sparse Estimation with Network Flow Optimization

Julien Mairal

University of California, Berkeley

Neyman seminar, Berkeley

Purpose of the talk

- introduce the literature on structured sparsity;
- introduce structured sparsity tools for graphs;
- solve the related combinatorial problems.

Acknowledgements

Bin
Yu

Francis
Bach

Rodolphe
Jenatton

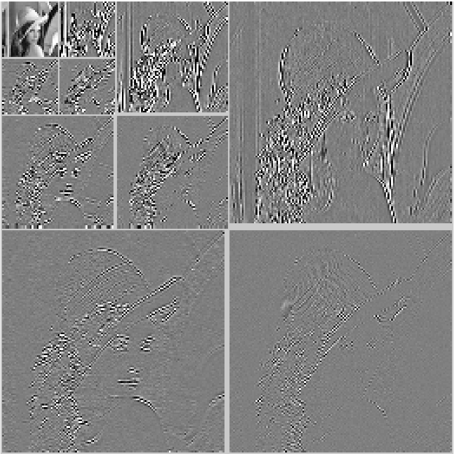
Guillaume
Obozinski

Laurent
Jacob

Part I: Introduction to Structured Sparsity

Wavelet coefficients

- Zero-tree wavelets coding [Shapiro, 1993];
- block thresholding [Cai, 1999].



Sparse linear models for natural image patches

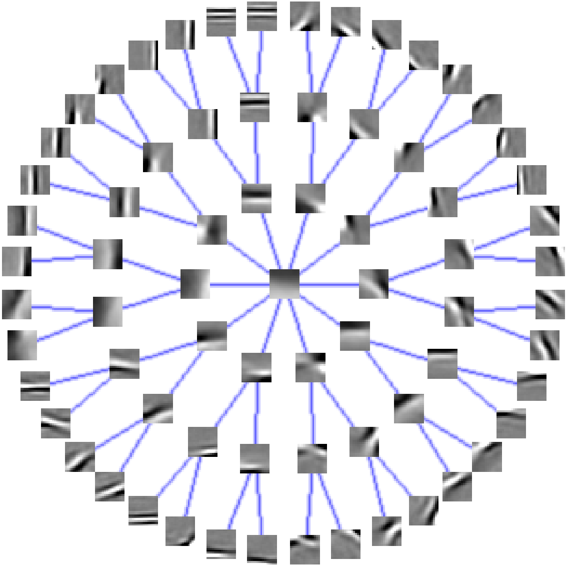
Image restoration

Sparse linear models for natural image patches

Image restoration

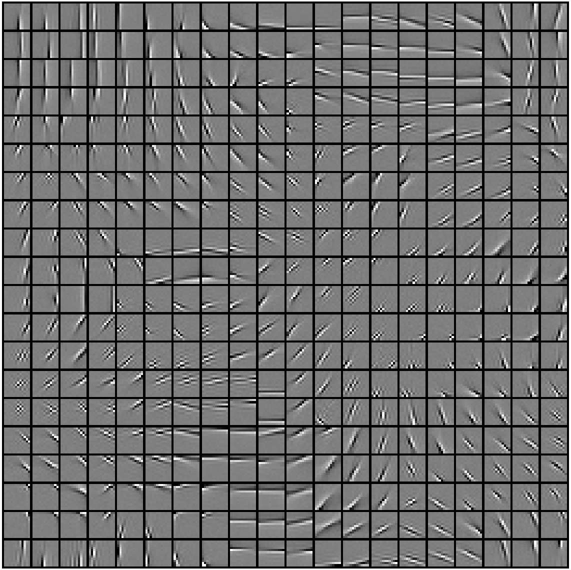
Structured dictionary for natural image patches

[Jenatton, Mairal, Obozinski, and Bach, 2010]



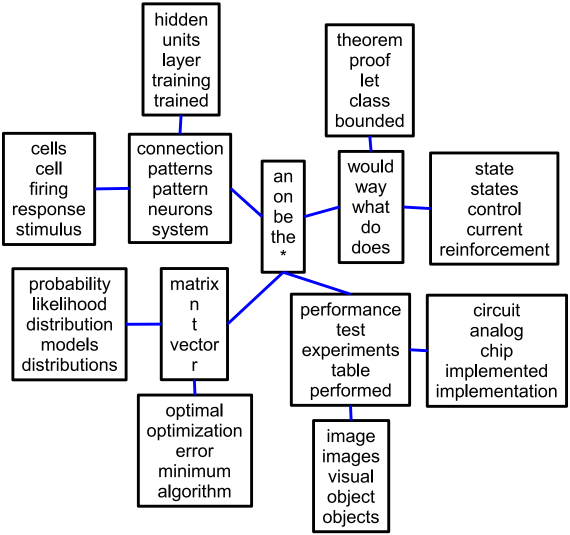
Structured dictionary for natural image patches

[Mairal, Jenatton, Obozinski, and Bach, 2011]



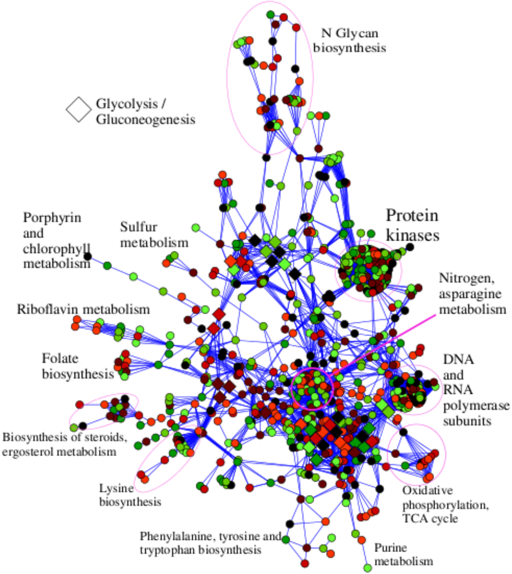
Tree of topics

[Jenatton, Mairal, Obozinski, and Bach, 2010]



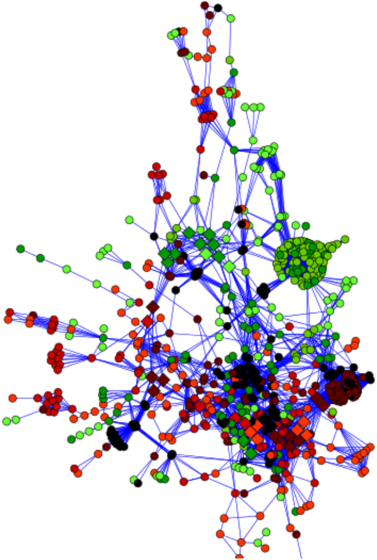
Metabolic network of the budding yeast

from Rapaport, Zinovyev, Dutreix, Barillot, and Vert [2007]



Metabolic network of the budding yeast

from Rapaport, Zinovyev, Dutreix, Barillot, and Vert [2007]



Questions about structured sparsity

$$\min_{\mathbf{w}\in\mathbb{R}^p} \underbrace{R(\mathbf{w})}_{\text{convex, smooth}} + \underbrace{\lambda\Omega(\mathbf{w})}_{\text{regularization}},$$

Ω should encode some a priori knowledge about $\mathbf{w}$.

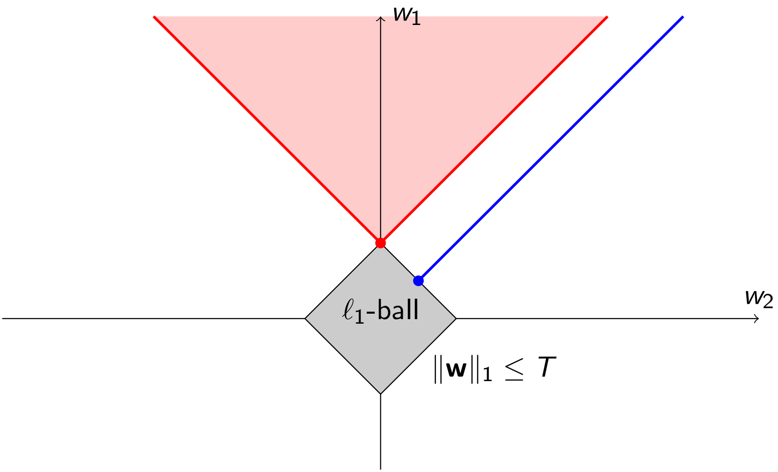
☺ In this talk, we will see

- how to design structured sparsity-inducing functions Ω;
- How to solve the corresponding estimation/inverse problems.

☹ out of the scope of this talk:

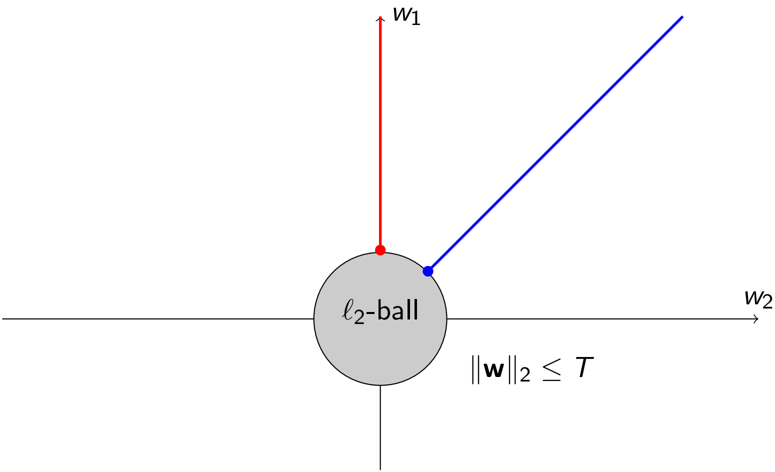
- consistency, recovery, theoretical properties.

Regularizing with the ℓ_1-norm



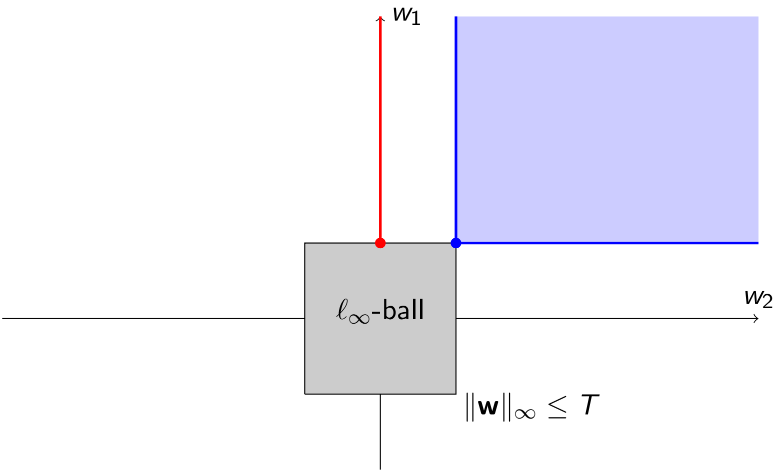
The projection onto a convex set is “biased” towards singularities.

Regularizing with the ℓ_2-norm



The ℓ_2-norm is isotropic.

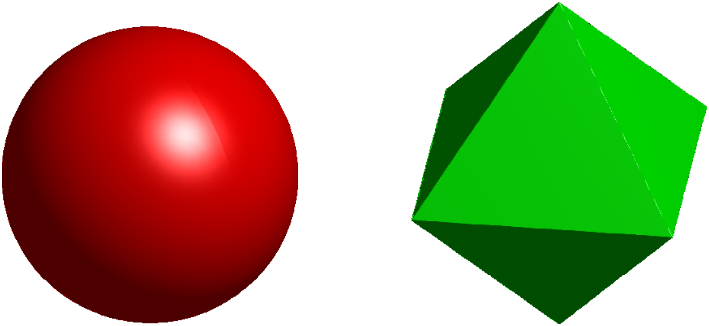
Regularizing with the ℓ_∞-norm



The ℓ_∞-norm encourages $|w_1| = |w_2|$.

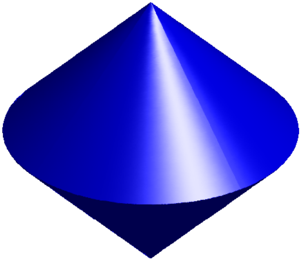
In 3D.

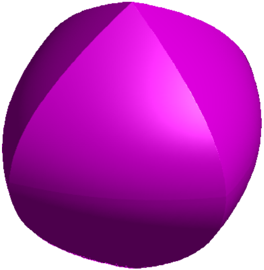
Copyright G. Obozinski



What about more complicated norms?

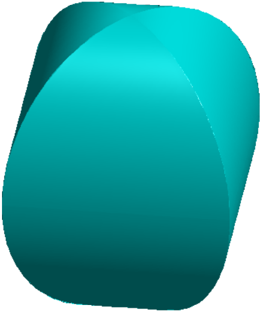
Copyright G. Obozinski

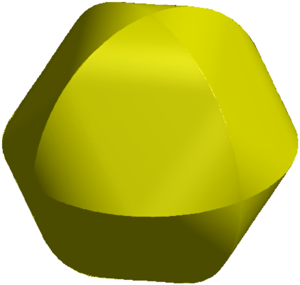




What about more complicated norms?

Copyright G. Obozinski



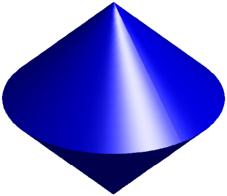


Group Lasso

Grandvalet and Canu [1999], Turlach et al. [2005], Yuan and Lin [2006]

$$\text{the } \ell_1/\ell_q\text{-norm :} \qquad \Omega(\mathbf{w}) = \sum_{g \in \mathcal{G}} \|\mathbf{w}_g\|_q.$$

- $\mathcal{G}$ is a **partition** of $\{1, \ldots, p\}$;
- $q = 2$ or $q = \infty$ in practice;
- can be interpreted as the ℓ_1-norm of $[\|\mathbf{w}_g\|_q]_{g \in \mathcal{G}}$.



$$\Omega(\mathbf{w}) = \|\mathbf{w}_{\{1,2\}}\|_2 + |\mathbf{w}_3|.$$

Structured sparsity with overlapping groups

Warning: Under the name "structured sparsity" appear in fact significantly different formulations!

1. non-convex
   - zero-tree wavelets [Shapiro, 1993];
   - predefined collection of sparsity patterns: [Baraniuk et al., 2010];
   - **select a union of groups: [Huang et al., 2009]**;
   - structure via Markov Random Fields: [Cehver et al., 2008];
2. convex (norms)
   - **tree-structure: [Zhao et al., 2009]**;
   - **select a union of groups: [Jacob et al., 2009]**;
   - **zero-pattern is a union of groups: [Jenatton et al., 2009]**;
   - other norms: [Micchelli et al., 2011].

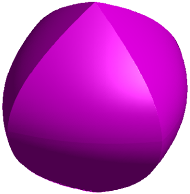
Group Lasso with overlapping groups

[Jenatton, Audibert, and Bach, 2009]

$$\Omega(\mathbf{w}) = \sum_{g \in \mathcal{G}} \|\mathbf{w}_g\|_q.$$

What happens when the groups overlap?

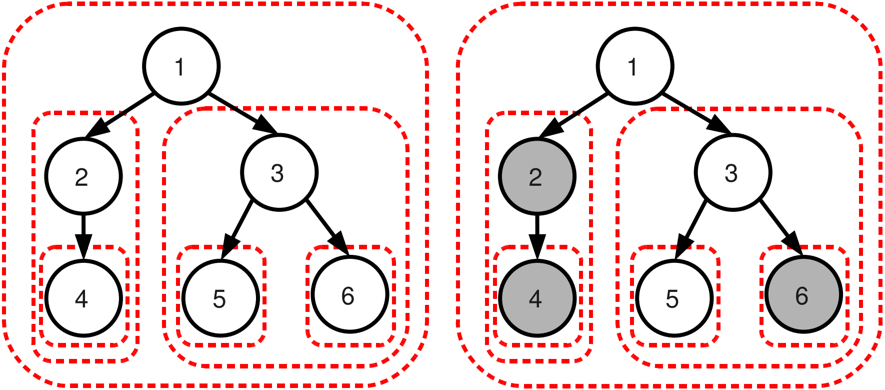
- the pattern of non-zero variables is an intersection of groups;
- the **zero pattern is a union of groups**.



$$\Omega(\mathbf{w}) = \|\mathbf{w}\|_2 + |\mathbf{w}_2| + |\mathbf{w}_3|.$$

Hierarchical Norms

[Zhao, Rocha, and Yu, 2009]



A node can be active only if its **ancestors are active**.

The selected patterns are **rooted subtrees.**

Modelling Patterns as Unions of Groups

the non-convex penalty of Huang, Zhang, and Metaxas [2009]

Warning: different point of view than the two previous slides

$$\varphi(\mathbf{w}) \triangleq \min_{\mathcal{J} \subseteq \mathcal{G}} \Big\{ \sum_{g \in \mathcal{J}} \eta_g \;\; \text{s.t.} \;\; \text{Supp}(\mathbf{w}) \subseteq \bigcup_{g \in \mathcal{J}} g \Big\}.$$

- the penalty is **non-convex**.
- is **NP-hard** to compute (set cover problem).
- The pattern of non-zeroes in $\mathbf{w}$ is a **union** of (a few) groups.

It can be rewritten as a boolean linear program:

$$\varphi(\mathbf{w}) = \min_{\mathbf{x} \in \{0,1\}^{|\mathcal{G}|}} \Big\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathbf{N}\mathbf{x} \geq \text{Supp}(\mathbf{w}) \Big\}.$$

Modelling Patterns as Unions of Groups

convex relaxation and the penalty of Jacob, Obozinski, and Vert [2009]

The penalty of Huang et al. [2009]:

$$\varphi(\mathbf{w}) = \min_{\mathbf{x} \in \{0,1\}^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathbf{N}\mathbf{x} \geq \text{Supp}(\mathbf{w}) \right\}.$$

A convex LP-relaxation:

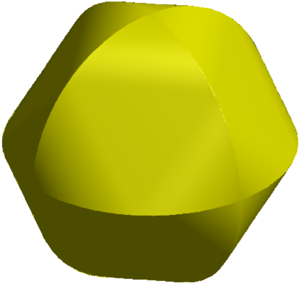
$$\psi(\mathbf{w}) \triangleq \min_{\mathbf{x} \in \mathbb{R}_+^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathbf{N}\mathbf{x} \geq |\mathbf{w}| \right\}.$$

Lemma: ψ is the penalty of Jacob et al. [2009] with the ℓ_∞-norm:

$$\psi(\mathbf{w}) = \min_{(\boldsymbol{\xi}^g \in \mathbb{R}^p)_{g \in \mathcal{G}}} \sum_{g \in \mathcal{G}} \eta_g \|\boldsymbol{\xi}^g\|_\infty \;\; \text{s.t.} \;\; \mathbf{w} = \sum_{g \in \mathcal{G}} \boldsymbol{\xi}^g \text{ and } \forall g, \; \text{Supp}(\boldsymbol{\xi}^g) \subseteq g,$$

Modelling Patterns as Unions of Groups

The norm of Jacob et al. [2009] in 3D



$\psi(\mathbf{w})$ with $\mathcal{G} = \{\{1,2\},\{2,3\},\{1,3\}\}$.

First-order/proximal methods

$$\min_{\mathbf{w}\in\mathbb{R}^p} \; R(\mathbf{w}) + \lambda\Omega(\mathbf{w})$$

- R is convex and differentiable with a Lipshitz gradient.
- Generalizes the idea of gradient descent

$$\mathbf{w}^{k+1} \leftarrow \arg\min_{\mathbf{w}\in\mathbb{R}^p} \underbrace{R(\mathbf{w}^k) + \nabla R(\mathbf{w}^k)^\top(\mathbf{w}-\mathbf{w}^k)}_{\text{linear approximation}} + \underbrace{\frac{L}{2}\|\mathbf{w}-\mathbf{w}^k\|_2^2}_{\text{quadratic term}} + \lambda\Omega(\mathbf{w})$$

$$\leftarrow \arg\min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{w} - (\mathbf{w}^k - \frac{1}{L}\nabla R(\mathbf{w}^k))\|_2^2 + \frac{\lambda}{L}\Omega(\mathbf{w})$$

When $\lambda = 0$, $\mathbf{w}^{k+1} \leftarrow \mathbf{w}^k - \frac{1}{L}\nabla R(\mathbf{w}^k)$, this is equivalent to a classical gradient descent step.

First-order/proximal methods

- They require solving efficiently the **proximal operator**

$$\min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{u}-\mathbf{w}\|_2^2 + \lambda\Omega(\mathbf{w})$$

- For the ℓ_1-norm, this amounts to a soft-thresholding:

$$\mathbf{w}_i^\star = \text{sign}(\mathbf{u}_i)(\mathbf{u}_i - \lambda)^+.$$

- There exists accelerated versions based on Nesterov optimal first-order method (gradient method with "extrapolation") [Beck and Teboulle, 2009, Nesterov, 2007, 1983];
- suited for large-scale experiments;
- can be used for non-convex optimization.

First-order/proximal methods

A few proximal operators:

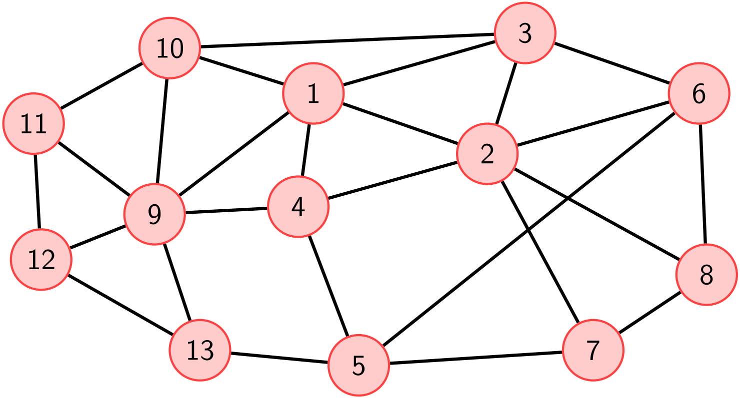
- ℓ_0-penalty: hard-thresholding;
- ℓ_1-norm: soft-thresholding;
- group-Lasso: group soft-thresholding;
- fused-lasso (1D total variation): [Hoefling, 2010];
- **hierarchical norms**: [Jenatton et al., 2010], $O(p)$ complexity;
- **overlapping group Lasso with ℓ_∞-norm**: [Mairal et al., 2010], (link with network flow optimization);

Part II: Structured Sparsity for Graphs

joint work with B. Yu

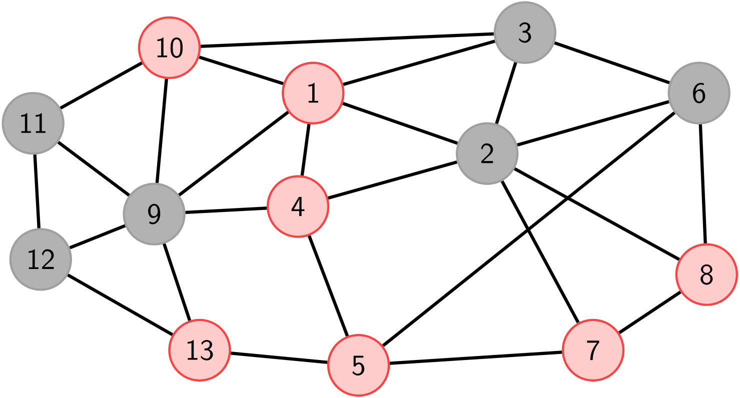
Graph sparsity

$G = (V, E)$, with $V = \{1, \ldots, p\}$



Graph sparsity

Encouraging patterns with a small number of connected components



Formulation

$$\min_{\mathbf{w}\in\mathbb{R}^p} \underbrace{R(\mathbf{w})}_{\text{convex, smooth}} + \underbrace{\lambda\Omega(\mathbf{w})}_{\text{regularization}},$$

Ω should encourage connected patterns in the graph.

- the penalty of Huang et al. [2009]:

$$\varphi(\mathbf{w}) = \min_{\mathbf{x}\in\{0,1\}^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathbf{N}\mathbf{x} \geq \text{Supp}(\mathbf{w}) \right\}.$$

- a convex LP-relaxation (penalty of Jacob et al. [2009]):

$$\psi(\mathbf{w}) \triangleq \min_{\mathbf{x}\in\mathbb{R}_+^{|\mathcal{G}|}} \left\{ \boldsymbol{\eta}^\top \mathbf{x} \;\; \text{s.t.} \;\; \mathbf{N}\mathbf{x} \geq |\mathbf{w}| \right\}.$$

Structured sparsity for graphs

Group structure for graphs.

Natural choices to encourage connectivity in the graph is to define $\mathcal{G}$ as

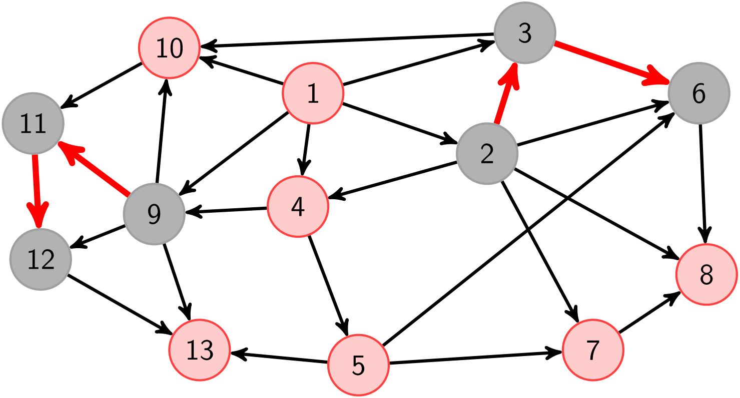
1. pairs of vertices linked by an arc. **only models local interactions**;
2. all connected subgraphs up to a size L. **cumbersome/intractable**;
3. all connected subgraphs. **intractable**.

Question

Can we replace connected subgraphs by another structure which (i) is rich enough to model long-range interactions in the graph, and (ii) leads to computationally feasible penalties?

Our solution when the graph is a DAG

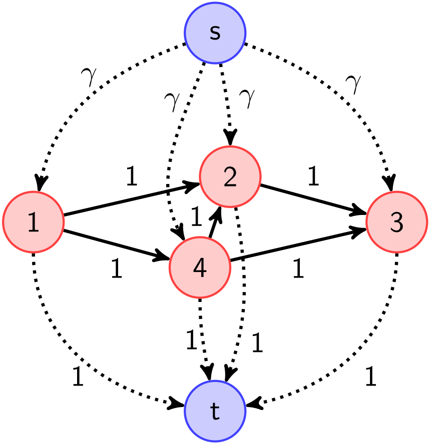
1. Define $\mathcal{G}$ to be the **set of all paths in the DAG**.
2. Define η_g to be $\gamma + |g|$ (the cost of selecting a path g).

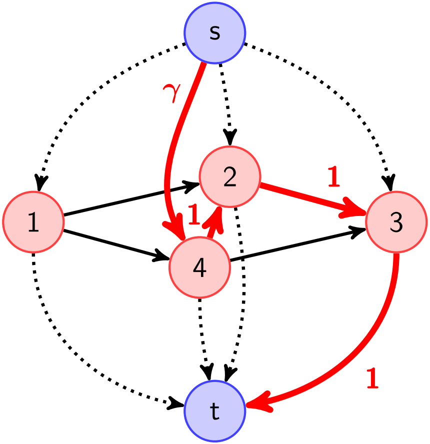


$$\varphi(\mathbf{w}) = (\gamma + 3) + (\gamma + 3)$$

Graph sparsity for DAGs

Decomposability of the weights $\eta_g = \gamma + |g|$



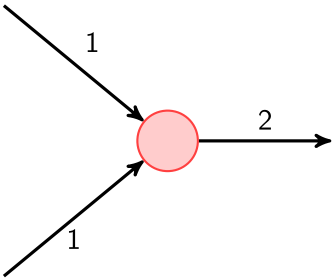


Quick introduction to network flows

References:

- Ahuja, Magnanti and Orlin. Network Flows, 1993
- Bertsekas. Network Optimization, 1998

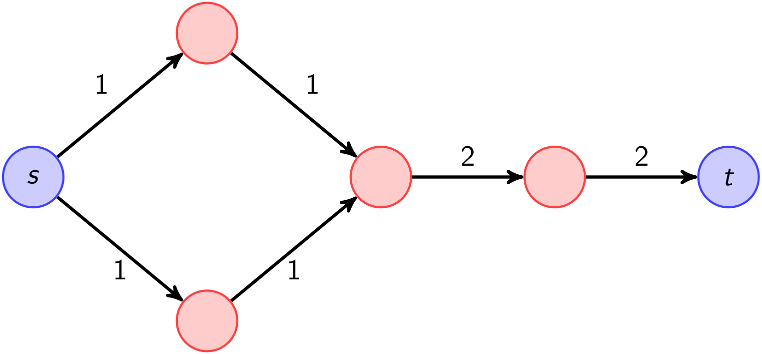
A flow f in $\mathcal{F}$ is a non-negative function on arcs that respects conservation constraints (Kirchhoff's law)



Quick introduction to network flows

Properties

Flows usually go from a source node s to a sink node t.



Quick introduction to network flows

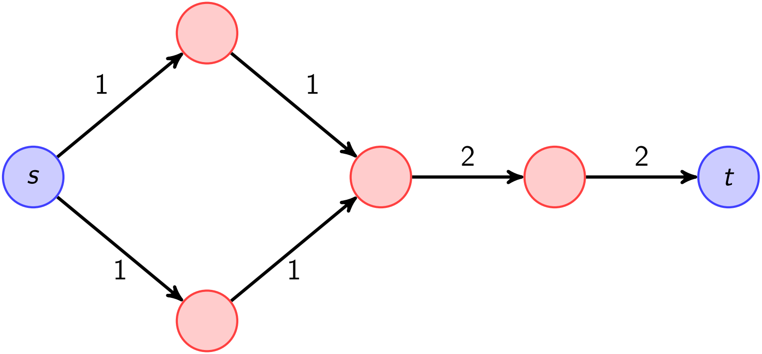
For a graph $G = (V, E)$:

- An arc (u, v) in E might have capacity constraints: $l_{uv} \leq f_{uv} \leq \delta_{uv}$.
- An arc (u, v) in E might have a cost: c_{uv}.
- Sending the maximum amount of flow in a network is called **maximum flow problem**.
- Finding a flow minimizing $\sum_{(u,v)\in E} f_{uv}c_{uv}$ is called **minimum cost flow problem**.
- These are **linear programs** with efficient dedicated algorithms [Goldberg, 1992] ($|V| = 100\,000$ is "fine").

Quick introduction to network flows

Properties

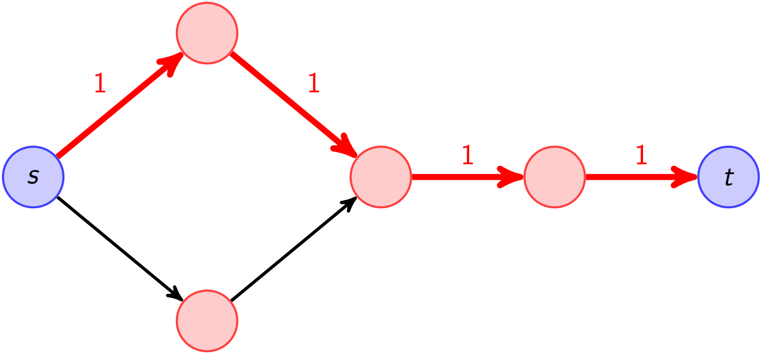
A flow on a DAG can be decomposed into "path-flows".



Quick introduction to network flows

Properties

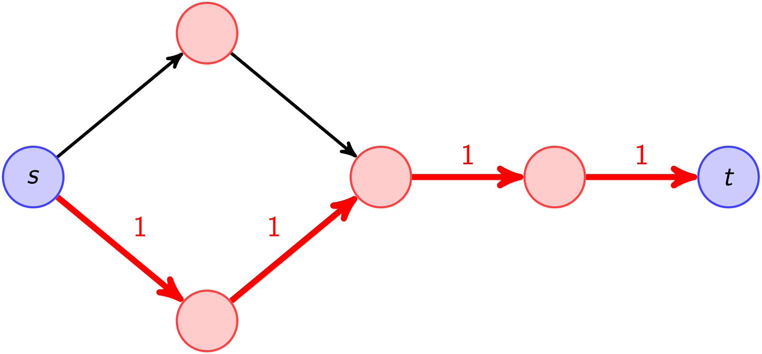
A flow on a DAG can be decomposed into "path-flows".



Quick introduction to network flows

Properties

A flow on a DAG can be decomposed into "path-flows".



Quick introduction to network flows

An optimization problem on paths might be transformed into an equivalent flow problem.

Proposition 1

$$\varphi(\mathbf{w}) = \min_{f \in \mathcal{F}} \sum_{(u,v)\in E'} f_{uv} c_{uv} \;\; \text{s.t.} \;\; s_j(f) \geq 1, \;\forall j \in \text{Supp}(\mathbf{w}),$$

Proposition 2

$$\psi(\mathbf{w}) = \min_{f \in \mathcal{F}} \sum_{(u,v)\in E'} f_{uv} c_{uv} \;\; \text{s.t.} \;\; s_j(f) \geq |\mathbf{w}_j|, \;\forall j \in \{1,\ldots,p\},$$

$\varphi(\mathbf{w})$, $\psi(\mathbf{w})$ and similarly the proximal operators, the dual norm of ψ **can be computed in polynomial time** using network flow optimization.

Application 1: Breast Cancer Data

The dataset is compiled from van't Veer et al. [2002] and the experiment follows Jacob et al. [2009].

Data description

- gene expression data of $p = 7910$ genes.
- $n = 295$ tumors, 78 metastatic, 217 non-metastatic.
- a graph between the genes was compiled by Chuang et al. [2007]. We arbitrary choose arc directions and heuristically remove cycles.

For each run, we keep 20% of the data as a test set, select parameters by 10-fold cross validation on the remaining 80% and retrain on 80%.

Application 1: Breast Cancer Data

Results

Results after 20 runs.

	Ridge	Lasso	Elastic-Net	Groups-pairs	ψ (convex)
error in %	31.0	36.0	31.5	35.9	30.2
error std.	6.1	6.5	6.7	6.8	6.8
nnz	7910	32.6	929	68.4	69.9
connex	58	30.9	355	13.1	1.3
stab	100	7.9	30.9	6.1	32

stab represents the percentage of genes selected in more than 10 runs.

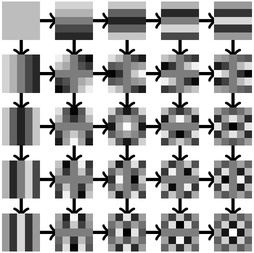
$\approx$ six proximal operators per second on our laptop cpu.

Application 2: Image denoising

Recipe, similarly to Elad and Aharon [2006]

- Extract all 10×10 overlapping patches from a noisy image.
- Obtain a sparse approximation of every patch.
- Average the estimates to obtain a clean image.

We use an orthogonal **DCT dictionary**:



Application 2: Image denoising

- Classical old-fashioned image processing dataset of 12 images.
- 7 levels of noise.
- Parameters optimized on the first 3 images.

σ	5	10	15	20	25	50	100
ℓ_0	**37.04**	33.15	31.03	29.59	28.48	25.26	22.44
ℓ_1	36.42	32.28	30.06	28.59	27.51	24.48	21.96
φ	37.01	**33.22**	**31.21**	**29.82**	**28.77**	**25.73**	**22.97**
ψ	36.32	32.17	29.99	28.54	27.49	24.54	22.12

PSNR: higher is better.

$\approx$ 4000 proximal operators per second on our laptop cpu.

Advertisement

- **Review monograph on sparse optimization:**
  F. Bach, R. Jenatton, J. Mairal and G. Obozinski. Optimization with Sparsity-Inducing Penalties. to appear in Foundation and Trends in Machine Learning.
- **SPAMS toolbox (C++)**
  - proximal gradient methods for ℓ_0, ℓ_1, elastic-net, fused-Lasso, group-Lasso, tree group-Lasso, tree-ℓ_0, sparse group Lasso, overlapping group Lasso...
  - ...for square, logistic, multi-class logistic loss functions.
  - handles sparse matrices, intercepts, provides duality gaps.
  - (block) coordinate descent, OMP, LARS-homotopy algorithms.
  - dictionary learning and matrix factorization (NMF).
  - fast projections onto some convex sets.
  - **soon: this work!**

Try it! `http://www.di.ens.fr/willow/SPAMS/`

References I

R. G. Baraniuk, V. Cevher, M. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 2010. to appear.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

T.T. Cai. Adaptive wavelet estimation: a block thresholding and oracle inequality approach. *Annals of Statistics*, 27(3):898–924, 1999.

V. Cehver, M. F. Duarte, C. Hegde, and R. G. Baraniuk. Sparse signal recovery usingmarkov random fields. In *Advances in Neural Information Processing Systems*, 2008.

H.Y. Chuang, E. Lee, Y.T. Liu, D. Lee, and T. Ideker. Network-based classification of breast cancer metastasis. *Molecular systems biology*, 3(1), 2007.

References II

M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 54(12):3736–3745, December 2006.

A. V. Goldberg. An Efficient Implementation of a Scaling Minimum-Cost Flow Algorithm. *Journal of Algorithms*, 22(1):1–29, 1992.

Y. Grandvalet and S. Canu. Outcomes of the equivalence of adaptive ridge with least absolute shrinkage. In *Advances in Neural Information Processing Systems*, 1999.

H. Hoefling. A path algorithm for the fused lasso signal approximator. *Journal of Computational and Graphical Statistics*, 19(4):984–1006, 2010.

J. Huang, Z. Zhang, and D. Metaxas. Learning with structured sparsity. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

References III

L. Jacob, G. Obozinski, and J.-P. Vert. Group Lasso with overlap and graph Lasso. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

R. Jenatton, J-Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. Technical report, 2009. preprint arXiv:0904.3523v1.

R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010.

J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Network flow algorithms for structured sparsity. In *Advances in Neural Information Processing Systems*, 2010.

J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Convex and network flow optimization for structured sparsity. *Journal of Machine Learning Research*, 12:2681–2720, 2011.

References IV

C.A. Micchelli, J.M. Morales, and M. Pontil. Regularizers for structured sparsity. *preprint arXiv:1010.0556v2*, 2011.

Y. Nesterov. A method for solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Math. Dokl.*, 27:372–376, 1983.

Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, CORE, 2007.

F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, and J.P. Vert. Classification of microarray data using gene networks. *BMC bioinformatics*, 8(1):35, 2007.

J.M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12): 3445–3462, 1993.

B. A. Turlach, W. N. Venables, and S. J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005.

References V

L.J. van't Veer, H. Dai, M.J. van de Vijver, Y.D. He, AA Hart, M. Mao, H.L. Peterse, K. van der Kooy, M.J. Marton, A.T. Witteveen, et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530, 2002.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B*, 68:49–67, 2006.

P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. 37(6A):3468–3497, 2009.